Novel SRAM Bias Control Circuits for a Low Power L1 Data Cache

Azam Seyedi[†‡] Adrià Armejach[†‡]
Adrián Cristal[†°] Osman S. Unsal[†] Mateo Valero[†]
[†]BSC-Microsoft Research Centre [‡]Universitat Politècnica de Catalunya
[°]IIIA - Artificial Intelligence Research Institute CSIC - Spanish National Research Council
{azam.seyedi, adria.armejach, adrian.cristal, osman.unsal, mateo.valero}@bsc.es

***Abstract*— This paper proposes two novel bias control circuits to manage the power consumption of inactive cache cells in data retention mode. Both circuits have lower power consumption and area overheads when compared to previous proposals. The first proposed circuit (Dynamic Bias Control circuit or DB-Control circuit) dynamically tracks the reference current and sets the bias voltage of cells, while the second (Self-Adjust Bias Control circuit or SAB-Control circuit) has a self-adjust property to set the bias voltages and also alleviates the instability problems that appear due to noise injection.**

Although any SRAM array can benefit from these circuits, to show their usefulness, we frame our study on a recently proposed dual-versioning L1 data cache that has been designed for chip multi-processors that implement optimistic concurrency proposals, where leakage current has more effect on power dissipation and on circuit instability. Therefore, we add the proposed bias control circuits to a 32KB dual-versioning SRAM (dvSRAM) cache and simulate and optimize the entire cache with 45-nm CMOS technology at 2GHz processor frequency and 1V supply voltage. The simulations demonstrate the effectiveness of our proposed circuits to reduce the energy consumption of dvSRAM L1 data cache by 35.8% on average compared to the typical dvSRAM cache. This is achieved with a modest area increase of 1.6% per sub-array and negligible delay overhead. We also show that instability problems are alleviated by using the SAB-Control circuit.

Keywords- Low Power Cache Design; Bias Control Circuit; dvSRAM; Optimistic Concurrency;

I. INTRODUCTION

On-die caches can consume more than 50% of the chip area in recent designs as technology scales [1]. With a 3× increase in leakage current on every technology generation, and with the dependence of the leakage power consumption on the number of transistors, caches will continue to account for a large component of leakage power dissipation in microprocessors [2]. Therefore, circuit techniques for controlling cache leakage are necessary for both high performance and low power consumption in nano-scale Large Scale Integrated (LSI) systems.

When a cache line is accessed in a typical cache, only a few parts of the whole cache get activated, and all un-selected rows just consume leakage power for data retention, so a leakage reduction technique is necessary to allow activating only small portions of the cache so that unused SRAM cells can be put into a low leakage state while retaining their data properly. This technique has been broadly called drowsy mode in prior works [3 … 11].

In this paper, we propose two novel bias control circuit techniques to reduce the power dissipation in drowsy mode. Our first proposed circuit, termed Dynamic Bias Control (DB-Control) circuit can dynamically track a reference current and produce an accurate cell bias voltage to reduce the power consumption. An external current source tracks and compensates any changes in the array´s leakage current; it enters each array and is transferred to a simple current to voltage converter circuit to produce the virtual VSS voltage level in drowsy mode.

Our second proposed circuit termed Self-Adjust Bias Control (SAB-Control) circuit sets the bias voltage to reduce the power consumption, and also has a self-adjust property to alleviate instability problems that might happen in drowsy mode. The SAB-Control circuit has two components, one generates virtual VDD, and the other generates virtual VSS; each consists of two diode-connected transistors in series with a resistor. By proper transistor sizing, the desired virtual VDD or virtual VSS level can be achieved; moreover, SAB-Control circuit adapts the voltage levels in the presence of noise injection and thus helps to reduce instability problems.

Our proposed bias control circuits are easy to implement and also have less overhead in terms of energy consumption and area compared to the recent designs [11]. In addition, SAB-Control circuit alleviates instability problems; this is important for drowsy caches, which usually tend to particularly suffer from instability problems due to noise [12].

We evaluate our proposed bias control circuits by adding them to a recently proposed 32KB recently proposed dual-versioning SRAM (dvSRAM) cache [13, 14], where active leakage current has more effect on energy consumption and instability problems. Then we optimize and simulate the entire cache. Our simulation results show that the energy consumption of the dual-versioning SRAM cache when using our proposed bias control circuits (dvSRAM-D) is, on average, 35.8% less than in the typical dvSRAM cache (dvSRAM-T). This is achieved by negligible delay overhead and reasonable area increase at, 1.6% per sub-array, about 50% lower compared to previous techniques which required around 3%

area increase per sub-array [11]. Although, the energy consumption of the added circuits is small, we compare the energy consumption overhead by replacing our proposed circuits by a previous reported technique [11] to control the cell bias voltages in dvSRAM cache. The results indicate that the average overhead energy in our proposed technique is 59% lower than the average overhead energy consumption in the technique reported in [11].

The rest of this paper is organized as follows: Section II summarizes related work in cache power reduction in drowsy mode. In Section III we introduce our novel bias control circuits. In Section IV we review dvSRAM main structure and its available operations and also we analyze the activation probability of dvSRAM cell components. In Section V we include the block diagram of a low power dvSRAM array when we add our proposed bias control circuits (dvSRAM-D). Section VI details the simulation results. In Section VII, we explain how our bias control circuit implementation alleviates aforementioned instability problem; and finally we conclude in Section VIII.

II. RELATED WORK IN CACHE POWER CONSUMPTION REDUCTION

During a typical cache access, relatively few cache circuit blocks are activated and un-active parts only dissipate leakage power. Many circuit techniques have been proposed to reduce the leakage power of un-selected cache parts. Some of them keep the un-active parts in sleep mode without attention to whether their data can be retained or not [3], while some of them define the drowsy mode for un-selected parts and can preserve the state of transistors in cache cells [4]. In drowsy mode techniques, the cell bias voltage is reduced and it is clamped at the level where the memory cell data can safely remain. There are many proposed drowsy mode techniques; one simple method is to use a sleep transistor to set VSS to a voltage higher than zero in a drowsy mode. In spite of simplicity, the sleep transistor needs to be sized properly to meet the wake-up timing requirement, to set the optimal VSS voltage level during the data-retention mode and to keep the IR droop low enough in the current path during active mode [5, 6].

Another technique is the use of a diode-connected transistor to control the VSS voltage level and clamp it to the desired voltage level in drowsy mode [7]. Even in some improved techniques proposed later, such as active replica cell technique [8] and source-line self-bias technique [9], the major shortcoming is not solved completely: the variation of V_{th} directly impacts the accuracy of the VSS voltage level in drowsy mode, and also the VSS voltage level is going to be around V_{th} of the clamped transistor which is too high in data retention mode. Programmable bias transistors technique is introduced to solve the above problems [10]. This technique has to be set for the worst case or maximum SRAM leakage current. In addition, aging can change the device characteristics and thus should be accounted as well. This makes programmable bias transistors technique less effective in reducing cache leakage power [11]. A sleep transistor design with an active feedback based on Opamp is also proposed to overcome the variation. The proposed configuration guarantees that the voltage magnitude of the VSS node is no greater than a V_{REF} [11, 6]. The benefit of this technique compared to the previous ones is that the standby power minimization is done by tracking and compensating any changes in leakage current. The major drawback of this scheme is the DC current power consumed by the Opamp which has to be replicated along each data bank to provide the needed granularity. In contrast, our proposed circuits have lower area overhead, lower energy consumption and settling time, and they also are easier to implement.

III. THE PROPOSED BIAS CONTROL CIRCUITS

In this section, we introduce our proposed circuits. Even though these circuits can be used in any component of the cache when in standby mode, we suggest, given their characteristics, to apply the DB-Control circuit to memory cells, which have to retain saved data while in drowsy mode; and the second circuit, SAB-Control circuit, which is even easier to implement, for the rest of the cache components.

A. DB-Control Circuit

To design this circuit, instead of using a reference voltage and operational amplifiers [11], we use a reference current and current to voltage converter circuits. Fig. 1 shows DB-Control circuit to generate and control the virtual VSS voltage level. The desired reference current is generated outside of the array and enters to a central current mirror circuit [15], which is located in the middle part of the array. The central current mirror circuit transfers this reference current to each sub-array via one intermediate current mirror circuit and a current to voltage converter circuit which its output node is connected to the virtual VSS rail of the sub-array. We use this intermediate current mirror circuit to increase the accuracy of our design because there is long distance between the central current mirror and the sub-array, so we cut the wire in the middle and add this current mirror [16].

The gate to source voltage of MN4 is lower than the gate to source voltage of MN1 because of considerable resistance and capacitance of long wires, so we use an accurate current mirror model proposed by Redouté and Steyaert [16]. If a typical current mirror structure [15] is used, equalizing the currents through MN1, MN4 and MN7 and also setting virtual VSS to the desired voltage level would be very time consuming. Transistor MN3 isolates the mirror node from the drain of MN1, transistor MN2 completes the DC biasing and finally transistors MN2 and MN3 fix the gate to source voltage of MN1 [16]. By optimizing the transistor sizes and considering resistance and capacitance of interconnections, the current I1 is equal to the current through MN1, IREF. A diode-connected PMOS Transistor (MP1) acts as a load for the intermediate current mirror circuit [15]. MN5 and MN6 act similarly than MN2 and MN3 to regulate the gate to source voltage of MN4. By proper transistor sizing and considering resistance and capacitance of interconnections I1 is equal to I2. The current to voltage converter consists of transistor MN7 and a diode-connected PMOS transistor, MP2, which converts the current I2 to the desired voltage level at the output node. Moreover, the diode connected PMOS transistor (MP2) tries to keep the virtual VSS voltage at the desired level. When the SRAM cells are not active, SW1 (SWR) is equal to zero (one) and transistor MN8 (MNS) is turned off (on); in active mode SW1 (SWR) is

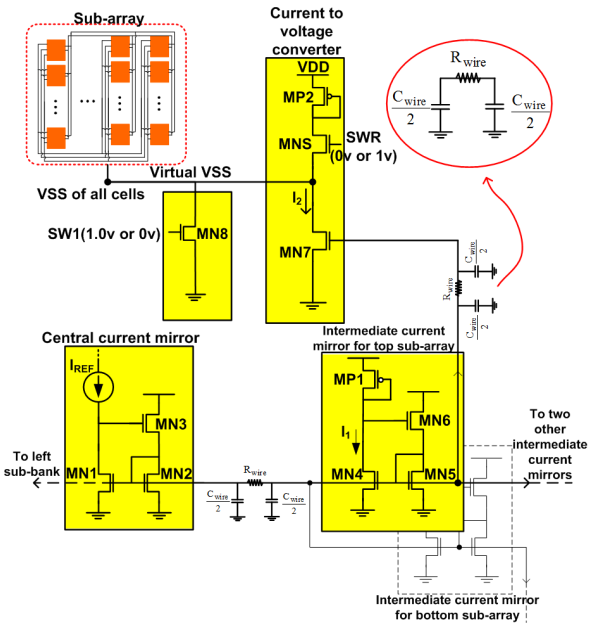

Figure 1: The proposed DB-Control circuit to generate and control virtual VSS of one sub-array. In the figure, a central current mirror circuit (located in the middle part of the array) and two intermediate current mirrors for the top and bottom sub-arrays (of the right sub-bank) are depicted; in addition, there is one current to voltage converter circuit for the top sub-array. Note that the central current mirror is also connected to two other intermediate current mirrors used by the rest of the sub-arrays of this sub-bank. Our cache structure is similar to dvSRAM cache structure [13].

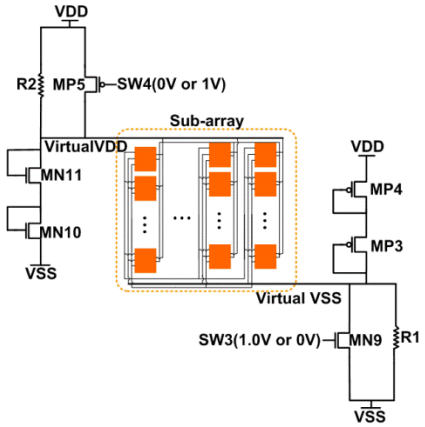

Figure 2: The proposed SAB-Control circuit to generate and control virtual VDD and virtual VSS for one sub-array.

high (low) and transistor MN8 (MNS) is turned on (off), and virtual VSS is discharged to zero.

B. *SAB-Control Circuit*

Now we introduce the SAB-Control circuit to generate and control virtual VDD and virtual VSS. Fig. 2 shows two bias control circuits, one for controlling virtual VDD and another for controlling virtual VSS. Virtual VSS is connected to VSS (Ground) via one resistor, R1, in parallel with a switch transistor, MN9, and also to VDD via a series of two diode-connected PMOS transistors, MP3 and MP4. A similar structure is used to generate virtual VDD; virtual VDD is connected to VSS via a series of two diode-connected NMOS transistors, MN10 and MN11 and also to VDD via one resistor, R2, in parallel with a switch transistor, MP5.

When the sub-array is accessed, SW3 is high, SW4 is low, and virtual VDD and virtual VSS are strongly connected to VDD and VSS respectively. When the sub-array is not accessed, SW3= 0V and SW4=1V, the virtual VSS rises to a voltage level equal to the voltage drop across R1 and also the virtual VDD is reduced to a voltage level lower than VDD as the magnitude of the voltage drop across R2. By proper sizing of transistors and resistors, we can clamp the virtual VDD and virtual VSS to the desired voltage levels in drowsy mode. So when the sub-array is not accessed, the bias voltage level is reduced which leads to a leakage power reduction. In Section VII, we explain how this bias control circuit can reduce the instability problems that might happen during drowsy mode.

IV. ANALYSIS OF THE DVSRAM L1 DATA CACHE

A. *Background : dvSRAM Cell Structure*

Researchers in [13, 14] recently proposed a new L1 data cache design able to manage two versions of the same logical data, useful for many well-known architectural optimistic concurrency implementations such as speculative multithreading [17], lock elision [18], or hardware transactional memory [19]. In this new cache design termed dual-versioning cache, each SRAM cell (dvSRAM) contains two typical standard 6T SRAM cells: the main cell and the secondary cell (Fig. 3). Each of them keeps a different version of the same logical data. These two cells are connected via two exchange circuits. When the exchange circuits are not active, the main cell and the secondary cell are isolated from each other. When dvStr is high, the data of the main cell is stored to the secondary cell, and when dvRstr is high, the data of the secondary cell is restored to the main cell. Table 1 shows dvSRAM cell operations [13].

B. *Profiling dvSRAM Components Activation*

As mentioned before, the circuit design of a dual-versioning L1 data cache based on dvSRAM cells targeting optimistic concurrency techniques was recently introduced [13]. The authors also report results for a Hardware Transactional Memory (HTM) use case [19], and the detailed microarchitecture implementation for the modified log-based HTM [14]. For our study, we use a similar setup to obtain the results shown in Table 2. The data indicates the number of operations of each kind done by the dvSRAM array for the four tested applications of the STAMP [20] benchmark suite, the state-of-the-art suite for HTM evaluation. It can be seen that the number of Read and Write operations is much higher compared to the others. Note that, for each operation, different parts of the selected dvSRAM cells are activated. For example, for the Read operation the main cells are activated, but for the dvBWrite operation, both the main and secondary cells are activated. Table 3 shows the active parts of dvSRAM cells for each operation. If it turns out that the probability of access to secondary cells is low, potentially large energy savings can be obtained from techniques that target secondary cells. Next, we calculate these probabilities.

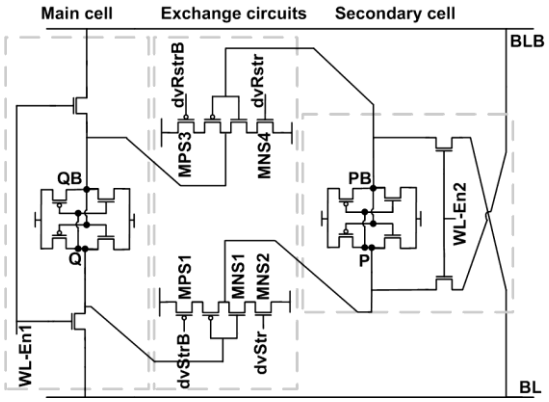

Figure 3: Circuit schematic of the dvSRAM cell. The main cell, the secondary cell and the exchange circuits are distinguished using dashed boxes [13].

Table 1: Brief description of the dvSRAM cell operations [13].

Operation	Description
Write	Writing to the main cell by activating WL-En1
Read	Reading from the main cell by activating WL-En1
dvWrite	Writing to the secondary cell by activating WL-En2
dvRead	Reading from the secondary cell by activating WL-En2
dvStr	Storing the main cell to the secondary cell by activating dvStr
dvRstr	Restoring the secondary cell to the main cell by activating dvRstr
dvBWrite	Writing to both cells simultaneously by activating WL-En1 and WL-En2
dvStrAll	Storing all main cells to their secondary cells simultaneously

The number of accesses to the main cells is calculated by adding up the total number of the following operations: Read, Write, dvBwrite, dvStr, dvRstr and dvStrAll. The number of accesses to the secondary cells is calculated by adding up the total number of the following operations: dvRead, dvWrite, dvBwrite, dvStr, dvRstr and dvStrAll, and the number of accesses to the exchange circuits is calculated by adding up the number of the following operations: dvStr, dvRstr, and dvStrAll. The probability of each part activation is then calculated by dividing the above numbers by the total number of operations for each tested benchmark as can be seen in Table 4. It is observed that the probability of main cells activation is much higher than the two others. On average, probability of activation for main, secondary and exchange circuits is 0.9999, 0.0637, and 0.0146 respectively. The secondary cells have to retain the data when accessing the main cells, which leads to substantial leakage power consumption; so if we want to use the benefits of dvSRAM cells in optimistic concurrency proposals, deploying some leakage control techniques that specifically target the secondary cells leakage current is vital.

V. DVSRAM-D ARRAY STRUCTURE DESIGN

In this section, we add our proposed bias control circuit to a typical dvSRAM array and show the block diagram of dvSRAM-D. As we explained in Section IV, for each operation different parts of dvSRAM cell are active so we exert different drowsy mode techniques for each part. The DB-Control circuit is useful for controlling bias voltage of cells (main and secondary cells), so we design two DB-Control circuits to generate virtual VSS (VSS1 and VSS2) and connect the VSS of all main cells and secondary cells of one sub-array to VSS1 and VSS2 respectively. SAB-Control circuit is useful for controlling bias voltage of exchange circuits, so we design a SAB-Control circuit to generate virtual VDD and VSS (VDD3 and VSS3), then we connect the VDD and VSS of all exchange circuits of one sub-array to nodes VDD3 and VSS3 respectively.

We consider that all the parts are in their drowsy mode unless related operations activate them. Main cells (secondary cells) are in their drowsy mode when none of them is active; at this time, VSS1 (VSS2) rises to the data retention voltage level that can be set with the DB-Control circuit. The exchange circuits are in their drowsy mode as well when all of them are inactive, but in this case, both of the VDD3 and VSS3 are set with the SAB-Control circuit.

Fig. 4 depicts the block diagram of a dvSRAM-D sub-array with data-in and data-out drivers, word-line address drivers and our proposed bias control circuits. The central current mirror circuits (designed for VSS1 and VSS2) are in the middle part of the array nearby decoders and control signal generators. The intermediate current mirror circuits are in the path to receive to the sub-array. Bias control circuits available for VDD3, VSS1, VSS2, and VSS3 generation can be seen in the upper and lower parts of the sub-array.

VI. EVALUATION

We simulate and optimize a 32KB typical dvSRAM array (dvSRAM-T) and a 32KB dvSRAM array including the proposed bias control circuits (dvSRAM-D) with Hspice 2003.03 using 45nm Predictive Technology Model [21], VDD=1V and T=25°C at the 2GHz processor clock frequency. We calculate the active energy per operation and static energy for both cases and show them in Table 5. For each operation, only the necessary parts of the SRAM cells are activated, and the other parts remain in drowsy mode as shown in Table 3. As can be seen in Table 5, the energy reduction for each operation depends on the number of active parts. It means our proposed techniques have a considerable effect on energy reduction for Read, Write, dvRead and dvWrite compared to other operations. Obviously, the energy consumption reduction for dvStr and dvRstr is less than other operations. Fig. 5 shows normalized energy consumption for the tested applications of the STAMP benchmark suite using dvSRAM-T and dvSRAM-D variants. The energy consumption reduction is 37.94% for Genome, 35.68% for Intruder, 35.09% for Kmeans, and 34.49% for Yada.

Fig. 6 shows the simulation waveforms of a random sequence of eleven operations. The simulation considers the first cell of the last row of a sub-array and takes a total of 15ns. The dvSRAM-D is designed to work at a two clock cycle access time (1ns), but an extra clock cycle is necessary for operations that adjust the signals VDD3, VSS1, VSS2, and VSS3 to change the mode. We calculate the average time to reach stable levels (low to high and high to low) as 0.34ns for VDD3, 0.47ns for VSS1, 0.45ns for VSS2 and 0.13ns for VSS3. It is important to note that operations that need an additional cycle are rare because the number of accesses that only use the main cells is much larger than the accesses that

Table 2: Number of each dvSRAM operation for tested applications of STAMP benchmark suite (Genome, Intruder, Kmeans, Yada) for one core.

	Read	Write	dvRead	dvBWrite	dvStr	dvRstr	dvStrAll
Genome	400449	25028	5	6873	0	103	406
Intruder	200258	31571	2	22546	0	13237	5089
Kmeans	1582689	34707	0	2092	0	27	530
Yada	711347	129899	46	142981	0	3461	634

Table 3: Activated parts of a dvSRAM cell for each operation.

Operation	Activated parts of a dvSRAM cells
Read	Main
Write	Main
dvRead	Secondary
dvWrite	Secondary
dvBWrite	Main, Secondary
dvStr	Main, Secondary, Exchange circuit
dvRstr	Main, Secondary, Exchange circuit
dvStrAll	Main, Secondary, Exchange circuit

Table 4: The probability of access to each part of dvSRAM cells for tested applications of the STAMP benchmark suite at each execution time.

	P(main)	P(secondary)	P(exchange)
Gnome	0.9999	0.0170	0.0011
Intruder	0.9999	0.1498	0.0672
Kmeans	1	0.0016	0.0003
Yada	0.9999	0.1488	0.0041

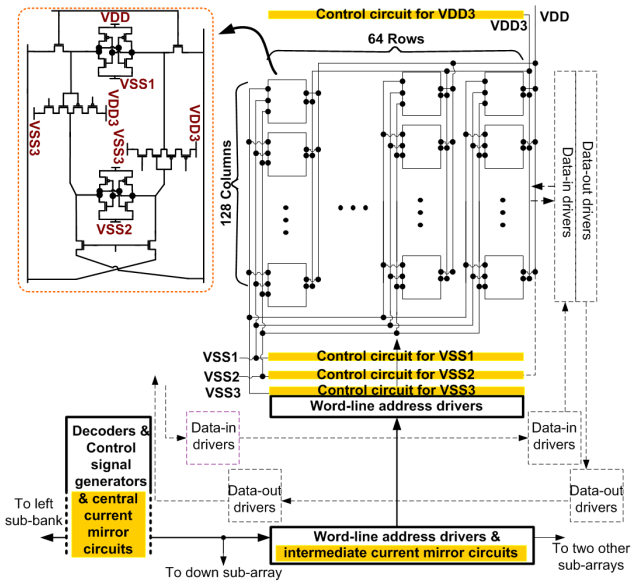

Figure 4: The block diagram of a dvSRAM-D sub-array. The proposed circuits are high-lighted.

Table 5: Energy consumption per operation for dvSRAM-T & dvSRAM-D.

Operation	Energy per operation (pJ/operation)	
	dvSRAM-T	dvSRAM-D
Read	112.4	59.1
Write	99.3	57.6
dvRead	114.4	63.1
dvWrite	101.1	59.0
dvBwrite	122.6	81.9
dvStr	114.9	84.6
dvRstr	112.9	84.7
dvStrAll	1123.2	819.9
Static	51.4	34.7

make use of the exchange circuits and/or secondary cells, which moreover usually happen in small bursts when executing speculatively for optimistic concurrency (in the case of HTM, this case happens inside a so-called "atomic" region). Thus, the timing overheads incurred by this additional cycle can be considered negligible, because as shown in Table 4, the probability of accessing to the main cells is dominant. We estimate the area overhead occupied by necessary bias control circuits [22] and add it to the total area. The area increase is 1.6% per sub-array, and it is lower than in previous techniques, which required around 3% increase per sub-array [11].

In addition, we construct for a dvSRAM-D Hspice transistor level net-list, but in this case, we replace the two DB-Control circuits that generate VSS1 and VSS2 with actively clamped sleep transistors [11] designed based on self-biased complementary folded cascade Opamps [23]. However, we do not replace the SAB-Control circuits that generate VDD3 and VSS3 in order to keep under control the instability problems. Although, the energy consumption of added circuits is small, we calculate the energy consumption of this setup for each operation and compare it with our proposal. The average overhead energy in our proposed technique is 59% lower than the average overhead energy consumption in the structure using actively clamped sleep transistors.

VII. INSTABILITY PROBLEMS OF UN-ACCESSED CELLS

In this section, we discuss an important instability problem that may occur in a dvSRAM array during execution time and show how our proposed SAB-Control circuit alleviates it. As mentioned in Section IV, the main cells are active much more often than the secondary cells, which have to retain their saved data. Main and secondary cells are connected to each other via exchange circuits; thus they are not completely separated, so we should pay attention to any undesired changes in the preserved data of secondary cells while they are un-accessed. This can be solved by a noise sensitive bias control circuit that we apply to dvSRAM array.

We explain the sensitivity problem with a simple example. Assume P=1 and that WL-En2 is low for some cycles in Fig. 3. When dvStr and dvRstr are low, the data is retained in the secondary cell. Also, assume Q=1 and that WL-En1 is low for some cycles too; hence the gate and the drain of transistor MNS1 are high. Any noise injection might increase the subthreshold leakage current of MNS2 and discharge node P and change the secondary and then the main cell values (remember WL-En1 and WL-En2 are low for some cycles). In the SAB-Control circuit, that additional leakage current of MNS2 increases virtual VSS (VSS3) voltage level (the voltage drop across R1), helping to reduce the additional leakage current that passes through MNS2 and decreases the probability of changing the saved data. Fig. 7 shows the

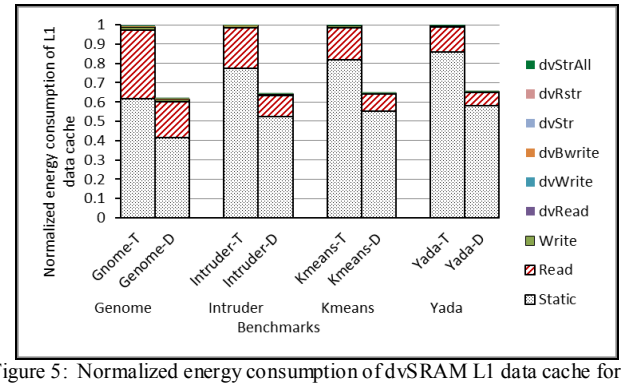

Figure 5: Normalized energy consumption of dvSRAM L1 data cache for tested applications of STAMP benchmark suite for dvSRAM-T & dvSRAM-D.

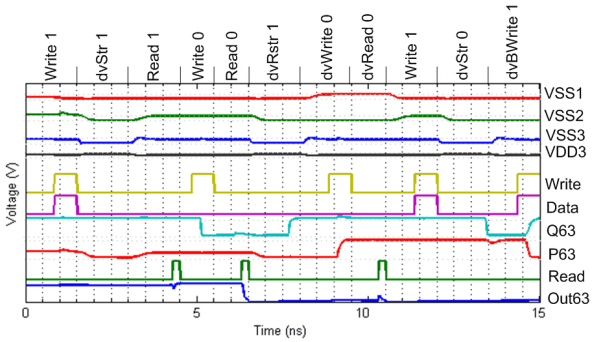

Figure 6: Simulation waveform for a sequence of eleven operations for the first cell of the last raw of a sub-array. The operations are shown on top. Max and min voltage levels for VSS1 (VSS2) are 0.3V and 0.05V, for VSS3 are 0.2V and 0.025V and for VDD3 are 0.910V and 0.85V respectively.

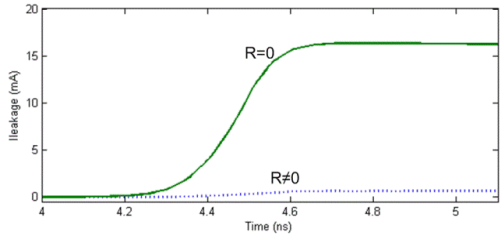

Figure 7: The leakage current passes through the transistors MNS1 and MNS2 when exchange circuits are un-accessed for R1=0 and R1≠0.

leakage current passing through transistors MNS1 and MNS2 in two modes; when R1=0 and when R1≠0. It demonstrates that R1 has a considerable effect in leakage current reduction. Similarly, when PB=0 and QB=0 for some cycles, our proposed SAB-Control circuit alleviates the instability problems.

VIII. CONCLUSION

In this paper, we propose two novel bias control circuits, DB-Control circuit and SAB-Control circuit, and apply them to a recently proposed dual-versioning L1 data cache to manage the energy consumption of un-selected SRAM cells. We present the design details and calculate the energy consumption for each operation and the total energy consumption for tested benchmarks. Our simulations demonstrate the usefulness of our proposed circuits to reduce the energy consumption with acceptable area increase and negligible delay overhead; we also show how our proposed techniques can reduce the instability problems that might appear, especially in drowsy mode. As an opportunity for future work, the DB-Control circuit could be applied to each cache line instead of the entire sub-array, which would lead to a finer granularity power management control. The DB-Control circuits designed for each cache line would have much smaller size than the one we propose for the entire sub-array, keeping the total area overhead low.

IX. REFERENCES

[1] J. Warnock et al. "A 5.2GHz microprocessor chip for the IBM zEnterprise™ system," ISSCC 2011.

[2] C. Kim, et al. " A Forward Body-Biased Low-Leakage SRAM Cache: Device, Circuit and Architecture Considerations," IEEE JSSC 2005.

[3] M. Powell et al. "Gated-Vdd: A Circuit Technique to Reduce Leakage in Deep-Submicron Cache Memories," ISLPED 2000.

[4] N. Kim et al. "Circuit and Microarchitectural Techniques for Reducing Cache Leakage Power," IEEE Transactions on VLSI 2004.

[5] A. Agarwal et al. "A single-Vt low-leakage gated-ground cache for deep submission," IEEE JSSCC 2003.

[6] K. Zhang et al. "Low-power SRAMs in Nanoscale CMOS technologies," IEEE Transactions on Electron Devices 200

[7] A. Bhavnagarwala et al. "A pico-joule class, 1 GHz, 32 KB * 64 b DSP SRAM with self reverse bias," IEEE Symposium on VLSI Circuits, Digest of Technical Papers 2003.

[8] Y. Takeyama et al. "A Low Leakage SRAM Macro with Replica Cell Biasing Scheme," IEEE JSSCC 2006.

[9] M. Yamaoka et al. "A 300-MHz 25uA/Mb-Leakage On-Chip SRAM Module Featuring Process-Variation Immunity and Low-Leakage-Active Mode for Mobile-Phone Application Processor," IEEE JSSCC 2005.

[10] K. Zhang et al. "SRAM design on 65-nm CMOS technology with dynamic sleep transistor for leakage reduction," IEEE JSSCC 2005.

[11] M. Khellah et al."A 256-Kb Dual-Vcc SRAM Building Block in 65-nm CMOS Process with Actively Clamped Sleep Transistor," JSSCC 2007.

[12] W.Wong et al."VOSCH: Voltage Scaled Cache Hierarchies", ICCD2007

[13] A. Seyedi et al. "Circuit Design of a Dual-Versioning L1 Data Cache for Optimistic," GLSVLSI 2011.

[14] A. Armejach et al."Using a Reconfigurable L1 Data Cache for Efficient Version Management in Hardware Transactional Memory," PACT 2011.

[15] B. Razavi,"Design of Analog CMOS Integrated Circuits" McGraw-Hill 2000.

[16] J. Redoute et al. "Current mirror structure insensitive to conducted EMI," Electronics Letters, 2005.

[17] V. Krishnan et al. "A Chip-Multiprocessor Architecture with Speculative Multithreading," IEEE Transactions on Computers 1999.

[18] R. Rajwar et al. "Speculative lock elision: Enabling highly concurrent multithreaded execution," MICRO 2001.

[19] T. Harris et al. "Transactional Memory, 2nd edition," Synthesis Lectures on Computer Architecture 2010.

[20] C. Cao Minh, et al. "STAMP: Stanford transactional applications for multi-processing," IISWC 2008.

[21] Predictive technology model: http://ptm.asu.edu/.

[22] H. Yoshida et al. "Accurate pre-layout estimation of standard cell characteristics," DAC 2004.

[23] B.G. Song, et al. "A 1.8 V self-biased complementary folded cascode amplifier," AP-ASIC 1999.